Data Analysis and Visualization in R (IN2339)

Exercise Session 2 - Data Wrangling

Daniela Klaproth-Andrade, Julien Gagneur

In this exercise session, we are analyzing an adapted version of a data set for book ratings, which contains 278,858 users (anonymized but with demographic information) providing 1,149,780 ratings about 271,379 books. We provide three different files containing information on the users, books and ratings. [https://www.kaggle.com/ruchi798/bookcrossing-dataset]

Section 00 - Getting ready

1. Make sure you have already installed and loaded the libraries `data.table` and `magrittr` by running the following commands:

```
install.packages("data.table")
install.packages("magrittr")
library(data.table)
library(magrittr)
```

Section 01 - Reading and cleaning up data

1. Load the three given datasets as `data.tables` and name them as `users_dt`, `books_dt` and `ratings_dt` accordingly. *Hint:* `fread()`

```
data_folder_name <- "./extdata/"

users_dt <- fread(paste0(data_folder_name, "BX-Users.csv"),
                  na.strings=c("NULL", "NA"), encoding = "UTF-8", sep = ",")
books_dt <- fread(paste0(data_folder_name, "BX-Books.csv"),
                  na.strings=c("NULL", "NA"), encoding = "UTF-8", sep = ",")
ratings_dt <- fread(paste0(data_folder_name, "BX-Book-Ratings.csv"),
                  na.strings=c("NULL", "NA"), encoding = "UTF-8", sep = ",")
```

2. Check the classes of `users_dt`, `ratings_dt` and `books_dt`. Confirm that these are indeed a `data.table`.

```
class(users_dt)
```

```
## [1] "data.table" "data.frame"
```

```
class(ratings_dt)
```

```
## [1] "data.table" "data.frame"
```

```
class(books_dt)
```

```
## [1] "data.table" "data.frame"
```

3. Check the column names and classes of the `users_dt` data table and change the type of the `Age` column in `users_dt` to numeric.

```
# Column names
colnames(users_dt)
```

```
## [1] "User-ID"  "Location" "Age"
```

```
# Column classes
sapply(users_dt, class)
```

```
##     User-ID    Location         Age
##   "integer" "character" "character"
```

```
## Change the type of Age to be numeric
users_dt[, Age := as.numeric(Age)]
```

```
## Warning in eval(jsub, SDenv, parent.frame()): NAs introduced by coercion
```

4. Produce a summary of the variables in `books_dt`.

```
summary(books_dt)
```

```
##      ISBN            Book-Title        Book-Author        Year-Of-Publication
##  Length:262500      Length:262500      Length:262500      Min.   :   0
##  Class :character   Class :character   Class :character   1st Qu.:1989
##  Mode  :character   Mode  :character   Mode  :character   Median :1995
##                                                           Mean   :1959
##                                                           3rd Qu.:2000
##                                                           Max.   :2050
##   Publisher          Image-URL-S        Image-URL-M        Image-URL-L
##  Length:262500      Length:262500      Length:262500      Length:262500
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
```

5. Return the first 5 and last 5 observations of the table `ratings_dt`.

```
ratings_dt
```

```
##                ISBN User-ID Book-Rating
##       1: 0000913154  171118           8
##       2: 0001010565   86123           0
##       3: 0001010565  209516           0
##       4: 0001046438   23902           9
##       5: 0001046713  196149           0
##      ---
## 1012374: B000234N76  264317           0
## 1012375: B000234NC6  100906           0
## 1012376: B00029DGGO  100088           0
## 1012377: B0002JV9PY  179791           0
## 1012378: B0002K6K8O  179791           0
##                                                                  Book-Title
##       1:            The Way Things Work: An Illustrated Encyclopedia of Technology
##       2:                                                          Mog's Christmas
##       3:                                                          Mog's Christmas
##       4:                                                                    Liar
```

```
##       5:                                                 Twopence to Cross the Mersey
##      ---
## 1012374:                                                                Falling Angels
## 1012375: It Must've Been Something I Ate: The Return of the Man Who Ate Everything
## 1012376:                                                  Good Wife Strikes Back, The
## 1012377:                                                        The Blockade Runners
## 1012378:                                                        The Underground City
##                          Book-Author Year-Of-Publication
##       1: C. van Amerongen (translator)                1967
##       2:                   Judith Kerr                1992
##       3:                   Judith Kerr                1992
##       4:                   Stephen Fry                   0
##       5:               Helen Forrester                1992
##      ---
## 1012374:               Tracy Chevalier                2001
## 1012375:           Jeffrey Steingarten                2002
## 1012376:              Elizabeth Buchan                   0
## 1012377:                   Jules Verne                   0
## 1012378:                   Jules Verne                   0
```

```
## running 'ratings_dt' for a data table automatically displays the following
# head(ratings_dt, n=5)
# tail(ratings_dt, n=5)
```

6. Replace all the - in column names by underscores _ in all three data tables. For example, `Book-Title` should be renamed to `Book_Title`. *Hint:* You can use the function `gsub()` that replaces pattern in a character string by a defined replacement. For example, for replacing `R` by `DataViz` in the following sentence `s` we use:

```
s <- 'R is fun'
gsub('R', 'DataViz', s)
```

```
## [1] "DataViz is fun"
```

```
colnames(users_dt) <- gsub("-", "_", colnames(users_dt))
colnames(books_dt) <- gsub("-", "_", colnames(books_dt))
colnames(ratings_dt) <- gsub("-", "_", colnames(ratings_dt))
```

7. Delete the columns `Image-URL-S`, `Image-URL-M` and `Image-URL-L` in the table `books_dt`.

```
books_dt[, c("Image_URL_S", "Image_URL_M", "Image_URL_L"):=NULL]
```

8. What is the first year of publication? What is the last one?

```
books_dt[, min(Year_Of_Publication)]
```

```
## [1] 0
```

```
books_dt[, max(Year_Of_Publication)]
```

```
## [1] 2050
```

9. Remove all the books published before 1900 and later than 2019 from `books_dt`.

```
books_dt <- books_dt[Year_Of_Publication >= 1900 & Year_Of_Publication <= 2019 ]
```

Section 02 - Data Exploration

1. How many different authors are included in the table `books_dt`?

```
books_dt[, uniqueN(Book_Author)]
```

```
## [1] 97170
```

```
# uniqueN() is the same as length(unique())
```

2. How many different authors are included for each year of publication between 2000 and 2010 in `books_dt`?

```
books_dt[Year_Of_Publication >= 2000 &
           Year_Of_Publication <= 2010, uniqueN(Book_Author),
         by=Year_Of_Publication][order(Year_Of_Publication)]
```

```
##    Year_Of_Publication    V1
## 1:                2000 12057
## 2:                2001 11818
## 3:                2002 11942
## 4:                2003  9913
## 5:                2004  4536
## 6:                2005    38
## 7:                2006     3
## 8:                2008     1
## 9:                2010     2
```

3. In how observations is the age information missing in the ratings table `users_dt`?

```
users_dt[is.na(Age), .N] # or users_dt[, sum(is.na(Age))]
```

```
## [1] 108027
```

4. Have a look at all locations from teenager users the table `users_dt`.

```
users_dt[Age<=19 & Age>=13, unique(Location)] %>% head
```

```
## [1] "stockton, california, usa"      "porto, v.n.gaia, portugal"
## [3] "melbourne, victoria, australia" "weston, ,"
## [5] "langhorne, pennsylvania, usa"   "cologne, nrw, germany"
```

5. What is the maximum rating value in the ratings table?

```
ratings_dt[, max(Book_Rating, na.rm=TRUE)]
```

```
## [1] 10
```

6. What is the most common rating value larger than 0?

```
ratings_dt[Book_Rating>0, .N, by=Book_Rating][N==max(N)]
```

```
##    Book_Rating     N
## 1:           8 89855
```

7. Which are the book identifiers (ISBN) with the highest ratings?

```
ratings_dt[Book_Rating == max(Book_Rating, na.rm=TRUE), "ISBN"] %>% head
```

```
##          ISBN
## 1: 0001360469
## 2: 0001374869
## 3: 0001821326
## 4: 0001845039
## 5: 0001857258
## 6: 0001900277
```

8. Sort the ratings table according to the rating value of each book in descending order. *Hint*: `order()`

```
# ratings_dt <- ratings_dt[order(-Book_Rating)]
# or
setorder(ratings_dt, -Book_Rating)
ratings_dt
```

```
##                ISBN User_ID Book_Rating
##       1: 0001360469   10067          10
##       2: 0001374869   10067          10
##       3: 0001821326  201017          10
##       4: 0001845039   56399          10
##       5: 0001857258  266867          10
##      ---
## 1012374: B000234N76  264317           0
## 1012375: B000234NC6  100906           0
## 1012376: B00029DGGO  100088           0
## 1012377: B0002JV9PY  179791           0
## 1012378: B0002K6K8O  179791           0
##                                                                  Book_Title
##       1:                                                       Babe Dressing
##       2:                                 Baby Plays (Collins Baby and Toddler Series)
##       3:                          Paddington at the Tower (A Paddington Picture Book)
##       4:                                                   The Moon of Gomrath
##       5:                                              Little Wolf's Book of Badness
##      ---
## 1012374:                                                         Falling Angels
## 1012375: It Must've Been Something I Ate: The Return of the Man Who Ate Everything
## 1012376:                                                 Good Wife Strikes Back, The
## 1012377:                                                   The Blockade Runners
## 1012378:                                                   The Underground City
##                  Book_Author Year_Of_Publication
##       1:       Mandy Stanley                1997
##       2:       Fiona Pragoff                1994
##       3:        Michael Bond                1976
##       4:         Alan Garner                1983
##       5:         Ian Whybrow                1999
##      ---
## 1012374:     Tracy Chevalier                2001
## 1012375: Jeffrey Steingarten                2002
## 1012376:    Elizabeth Buchan                   0
## 1012377:         Jules Verne                   0
## 1012378:         Jules Verne                   0
```

9. Create a new column `Country` in the table `users_dt` for the name of the country of each user. For instance, from the location `cologne, nrw, germany`, we can assume the user comes from `Germany`. *Hint:* `tstrsplit()`

```
users_dt[, Country := tstrsplit(Location, ',')[[3]]]
users_dt
```

```
##         User_ID                                Location Age         Country
##      1:       1                   nyc, new york, usa  NA             usa
##      2:       2              stockton, california, usa  18             usa
##      3:       3        moscow, yukon territory, russia  NA          russia
##      4:       4              porto, v.n.gaia, portugal  17        portugal
##      5:       5     farnborough, hants, united kingdom  NA  united kingdom
##     ---
## 269057:  278107         kentville, nova scotia, canada  51          canada
## 269058:  278108 christchurch, new zealand, new zealand  18     new zealand
## 269059:  278109             malvern, pennsylvania, usa  16             usa
## 269060:  278110          peterborough, ontario, canada  35          canada
## 269061:  278111            grand rapids, michigan, usa  NA             usa
```

10. How many different countries are contained in the table `users_dt`?

```
users_dt[, uniqueN(Country)]
```

```
## [1] 946
```

11. What is the average age of the users in `users_dt`? What is the average age for users in NYC, Stockton and Moscow? *Hint:* use `by:=` and `i` for row filtering

```
# Overall average
users_dt[, .(mean_age = mean(Age, na.rm=TRUE))]
```

```
##    mean_age
## 1:  34.9379
```

```
#Find the average age of the users in the specified cities
# Create city column
users_dt[, City := tstrsplit(Location, ',')[[1]]]
# Compute average per city
users_dt[ City %in% c("nyc", "stockton", "moscow"), .(mean_age = mean(Age, na.rm=TRUE)),
          by=City]
```

```
##        City mean_age
## 1:      nyc 31.62500
## 2: stockton 38.49231
## 3:   moscow 27.80000
```

Section 03 - Manipulating data tables

1. Add a new column called `High_Rating` to the data table `ratings_dt`. The column has an integer 1 for all observations with a rating value higher than 7.

```
ratings_dt[, High_Rating := ifelse(Book_Rating > 7, 1, 0)]
```

2. How many observations are considered to be a high ranking? What is the proportion of high ranked observations among all observations?

```
ratings_dt[, sum(High_Rating)] # absolute
```

```
## [1] 219361
```

```r
ratings_dt[, sum(High_Rating)/.N] # relative
```

```
## [1] 0.2166789
```

3. Set the book identifier the key of the data table `books_dt`. What happened to the order of the data table? *Hint*: `setkey()`

```r
books_dt # Before setting key
```

```
##                ISBN
##      1: 0195153448
##      2: 0002005018
##      3: 0060973129
##      4: 0374157065
##      5: 0393045218
##     ---
## 257872: 0441297528
## 257873: 0441792642
## 257874: 0448164884
## 257875: 0451520521
## 257876: 0486234045
##                                                                                                   Book_
##      1:                                                                                         Classical Myth
##      2:                                                                                                Clara C
##      3:                                                                                        Decision in Nor
##      4: Flu: The Story of the Great Influenza Pandemic of 1918 and the Search for the Virus That Caus
##      5:                                                                                      The Mummies of Ur
##     ---
## 257872:                                                                                         The Golden Nag
## 257873:                                                                                     Sensei II: Sword M
## 257874:                                                                              More Stan Fishchler's Sports Stu
## 257875:                                                                                                     Ba
## 257876:          Sports picture quiz book: With 240 photographs from Photoworld, a division of F.P.G
##                Book_Author Year_Of_Publication                   Publisher
##      1:   Mark P. O. Morford                2002    Oxford University Press
##      2: Richard Bruce Wright                2001      HarperFlamingo Canada
##      3:         Carlo D'Este                1991            HarperPerennial
##      4:     Gina Bari Kolata                1999       Farrar Straus Giroux
##      5:      E. J. W. Barber                1999 W. W. Norton &amp; Company
##     ---
## 257872:     Jesica Salmonson                1982                 Ace Books
## 257873:        David Charney                1984               ACE Charter
## 257874:        Stan Fischler                1979     Berkley Pub Group (Mm)
## 257875:       Sinclair Lewis                1982               Signet Book
## 257876:         John Grafton                1977         Dover Publications
```

```r
setkey(books_dt, "ISBN")
books_dt # After setting key
```

```
##                ISBN
##      1: 0000913154
##      2: 0001010565
##      3: 0001046713
##      4: 000104687X
##      5: 0001046934
##     ---
```

```
## 257872: B0001PBXMS
## 257873: B0001PIOX4
## 257874: B000234N3A
## 257875: B000234N76
## 257876: B000234NC6
##                                                                  Book_Title
##      1:            The Way Things Work: An Illustrated Encyclopedia of Technology
##      2:                                                        Mog's Christmas
##      3:                                        Twopence to Cross the Mersey
##      4:                T.S. Eliot Reading \\""The Wasteland\\"" and Other Poems
##      5:                                        The Prime of Miss Jean Brodie
##     ---
## 257872:                                                            Love, etc.
## 257873:                                                         Fahrenheit 451
## 257874:                                                                  Fraud
## 257875:                                                         Falling Angels
## 257876: It Must've Been Something I Ate: The Return of the Man Who Ate Everything
##                         Book_Author Year_Of_Publication
##      1: C. van Amerongen (translator)                1967
##      2:                   Judith Kerr                1992
##      3:               Helen Forrester                1992
##      4:                    T.S. Eliot                1993
##      5:                  Muriel Spark                1999
##     ---
## 257872:                Julian Barnes                2001
## 257873:                 Ray Bradbury                1993
## 257874:                 David Rakoff                2001
## 257875:              Tracy Chevalier                2001
## 257876:           Jeffrey Steingarten                2002
##                         Publisher
##      1:        Simon &amp;amp; Schuster
##      2:                      Collins
##      3:    HarperCollins Publishers
##      4:    HarperCollins Publishers
##      5: Trafalgar Square Publishing
##     ---
## 257872:                       Knopf
## 257873:        Simon &amp;amp; Schuster
## 257874:                   Doubleday
## 257875:                   E P Dutton
## 257876:                       Knopf
```

```
# After setting the key the table is reordered in ascending order w.r.t the defined key
```

4. Which users did not give any rating to any book? Filter these users out from `users_dt`. *Hint*: There's no need to merge `users_dt` with `ratings_dt`, we are simply interested in the users that are not in `ratings_dt`.

```
users_who_rated <- ratings_dt[,User_ID]
users_dt[!User_ID %in%users_who_rated]
```

```
##         User_ID                              Location Age         Country
##      1:       1                    nyc, new york, usa  NA             usa
##      2:       3       moscow, yukon territory, russia  NA          russia
##      3:       4             porto, v.n.gaia, portugal  17        portugal
##      4:       5    farnborough, hants, united kingdom  NA  united kingdom
```

```
##      5:       6          santa monica, california, usa  61             usa
##     ---
## 180938:  278102              innsbruck, tirol, austria  34         austria
## 180939:  278103               baltimore, maryland, usa  46             usa
## 180940:  278105                 marseille, n/a, france  23          france
## 180941:  278108 christchurch, new zealand, new zealand  18     new zealand
## 180942:  278109             malvern, pennsylvania, usa  16             usa
##                 City
##      1:          nyc
##      2:       moscow
##      3:        porto
##      4:  farnborough
##      5: santa monica
##     ---
## 180938:    innsbruck
## 180939:    baltimore
## 180940:    marseille
## 180941: christchurch
## 180942:      malvern
```

5. What is the most common age of users who rated at least one book?

```
users_dt[User_ID%in%users_who_rated & !is.na(Age), .N, by=Age][N==max(N)]
```

```
##    Age    N
## 1:  26 1558
```

6. On average, how many books did a user rate?

```
ratings_dt[, .N, by=User_ID][, mean(N, na.rm=TRUE)]
```

```
## [1] 11.2414
```

7. What is the title of the first published book with the highest ranking?

```
ratings_dt[order(Year_Of_Publication, -Book_Rating),
           .(Book_Title, Year_Of_Publication, Book_Rating)] %>% head(1)
```

```
##       Book_Title Year_Of_Publication Book_Rating
## 1: Darcys Utopia                   0          10
```

8. In which year was a book with the largest number of ratings last published?

```
ratings_dt[, Rating_Count:=.N, by=ISBN]
ratings_dt[ Rating_Count == max(Rating_Count), max(Year_Of_Publication)]
```

```
## [1] 2004
```

9. Add to the table `ratings_dt` the highest ranking that each book received as a new column called `Max_Book_Ranking`.

```
ratings_dt[, Max_Book_Rating := max(Book_Rating), by=ISBN]
ratings_dt
```

```
##                ISBN User_ID Book_Rating
##       1: 0001360469   10067          10
##       2: 0001374869   10067          10
##       3: 0001821326  201017          10
##       4: 0001845039   56399          10
##       5: 0001857258  266867          10
```

```
##      ---
## 1012374: B000234N76  264317           0
## 1012375: B000234NC6  100906           0
## 1012376: B00029DGGO  100088           0
## 1012377: B0002JV9PY  179791           0
## 1012378: B0002K6K8O  179791           0
##                                                                    Book_Title
##       1:                                                       Babe Dressing
##       2:                                 Baby Plays (Collins Baby and Toddler Series)
##       3:                          Paddington at the Tower (A Paddington Picture Book)
##       4:                                                   The Moon of Gomrath
##       5:                                              Little Wolf's Book of Badness
##      ---
## 1012374:                                                         Falling Angels
## 1012375: It Must've Been Something I Ate: The Return of the Man Who Ate Everything
## 1012376:                                                 Good Wife Strikes Back, The
## 1012377:                                                       The Blockade Runners
## 1012378:                                                       The Underground City
##                  Book_Author Year_Of_Publication High_Rating Rating_Count
##       1:       Mandy Stanley                1997           1            1
##       2:       Fiona Pragoff                1994           1            1
##       3:        Michael Bond                1976           1            1
##       4:         Alan Garner                1983           1            1
##       5:         Ian Whybrow                1999           1            1
##      ---
## 1012374:     Tracy Chevalier                2001           0            1
## 1012375: Jeffrey Steingarten                2002           0            1
## 1012376:    Elizabeth Buchan                   0           0            1
## 1012377:         Jules Verne                   0           0            1
## 1012378:         Jules Verne                   0           0            1
##          Max_Book_Rating
##       1:              10
##       2:              10
##       3:              10
##       4:              10
##       5:              10
##      ---
## 1012374:               0
## 1012375:               0
## 1012376:               0
## 1012377:               0
## 1012378:               0
```

10. Subset the merged ratings table to contain only books written by the following authors:

```
authors <- c("Agatha Christie", "William Shakespeare", "Stephen King",
             "Ann M. Martin", "Carolyn Keene", "Francine Pascal",
             "Isaac Asimov", "Nora Roberts", "Barbara Cartland", "Charles Dickens")
```

How many ratings has each author? What is their max and average ranking?

```
ratings_dt_sub <- ratings_dt[Book_Author%in%authors]
ratings_dt_sub[, .(mean(Book_Rating), max(Book_Rating), .N), by=Book_Author]
```

```
##             Book_Author       V1 V2    N
##  1: William Shakespeare 3.956572 10 1727
```

```
##  2:     Agatha Christie 2.855744 10 2246
##  3:       Carolyn Keene 2.465116 10 1075
##  4:        Isaac Asimov 3.183565 10  937
##  5:        Stephen King 3.600363 10 9914
##  6:     Charles Dickens 2.735065 10 1155
##  7:        Nora Roberts 2.656009 10 8413
##  8:    Barbara Cartland 4.241176 10  340
##  9:     Francine Pascal 1.169464 10 1251
## 10:       Ann M. Martin 0.996904 10 1938
```